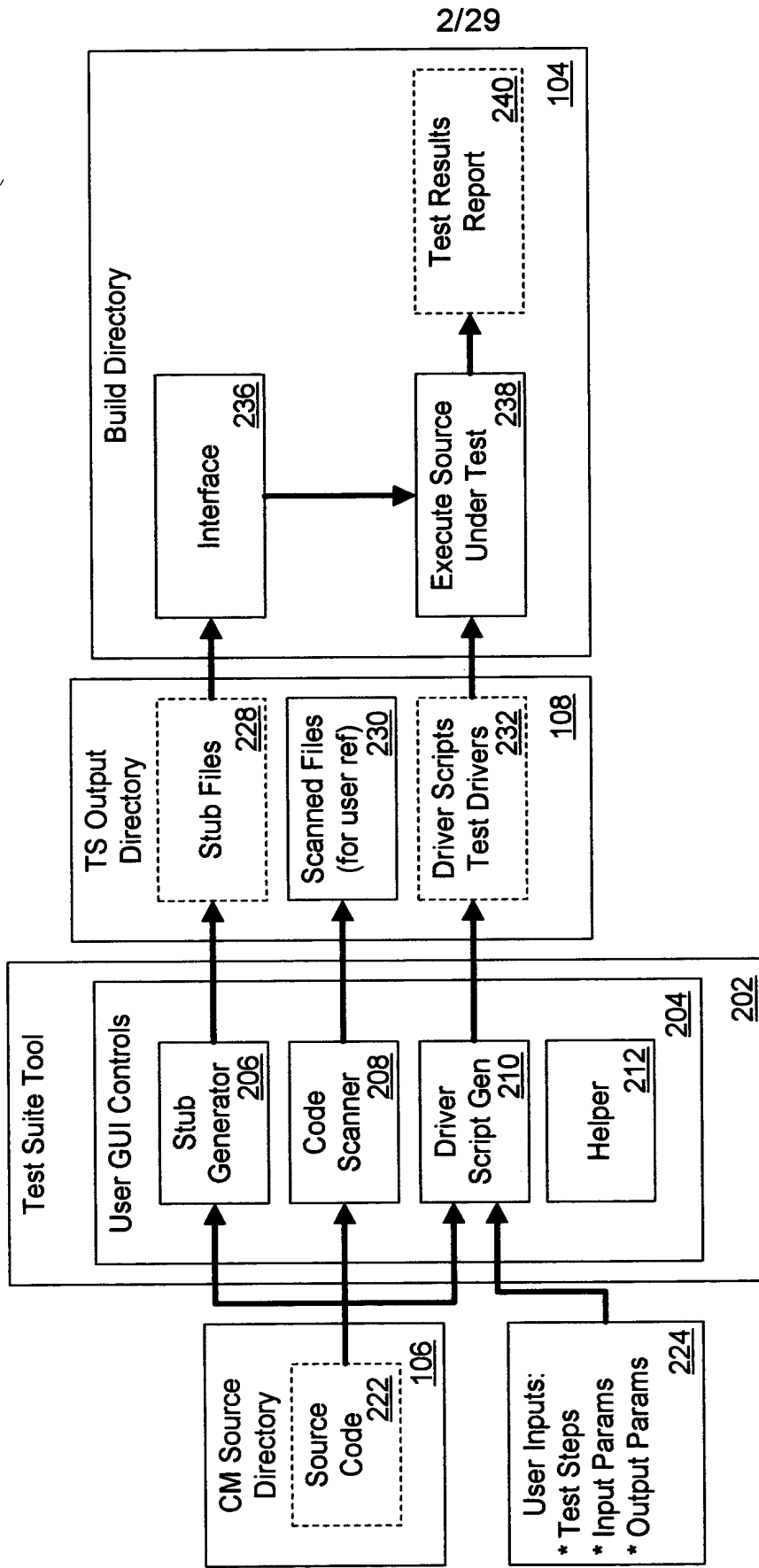


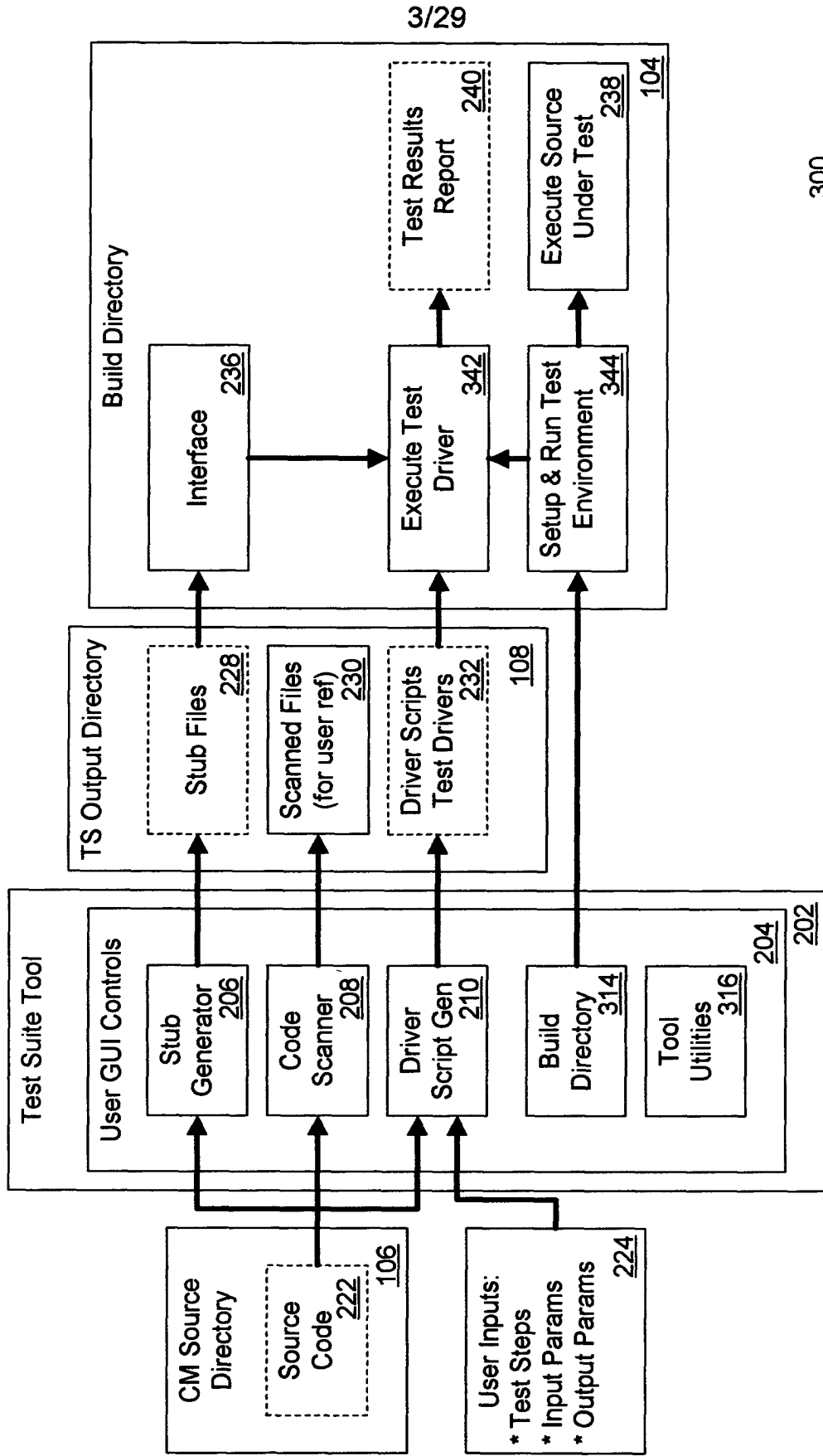
FIG. 1



2/29

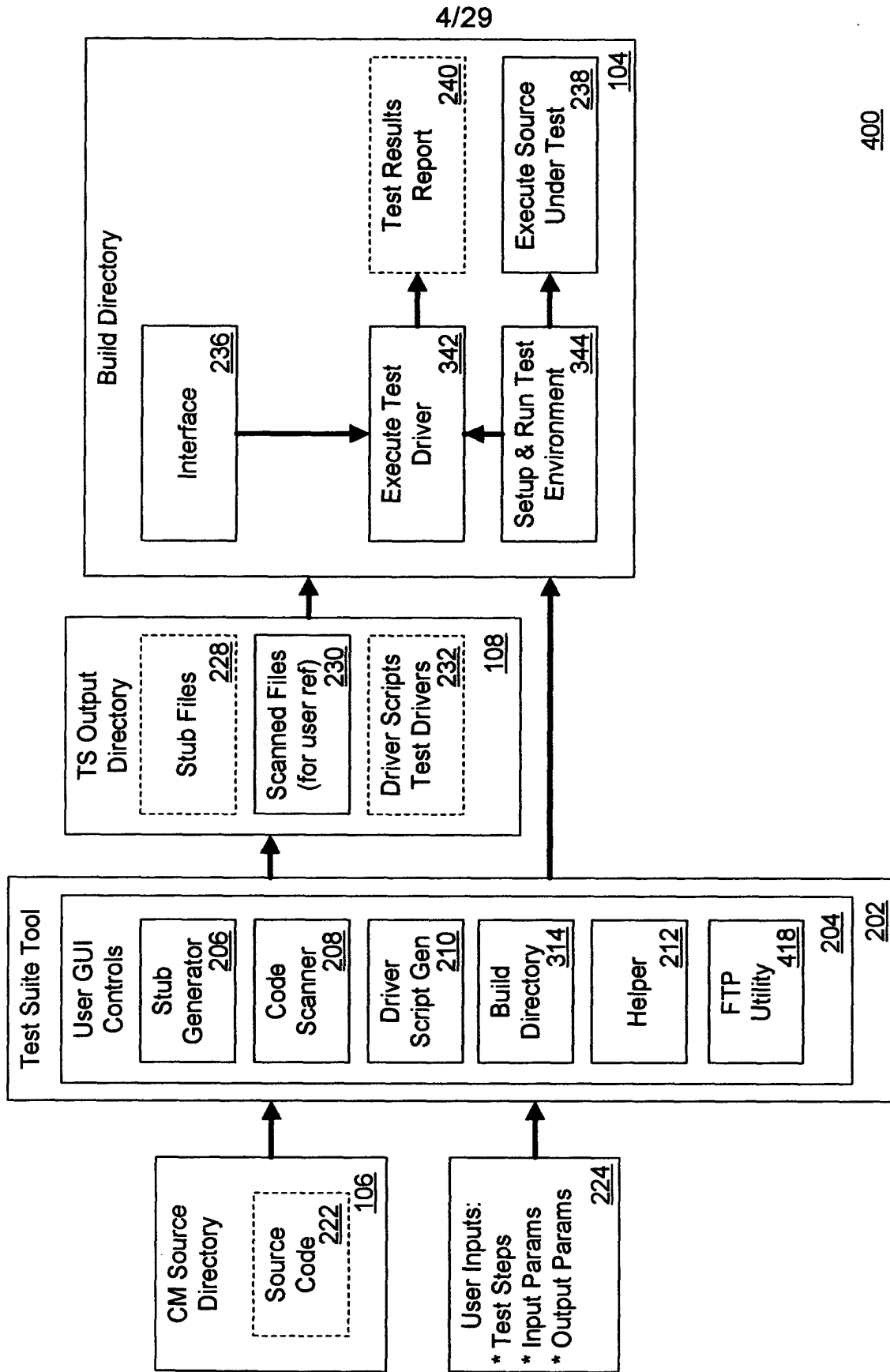
200

FIG. 2



300

FIG. 3



400

FIG. 4

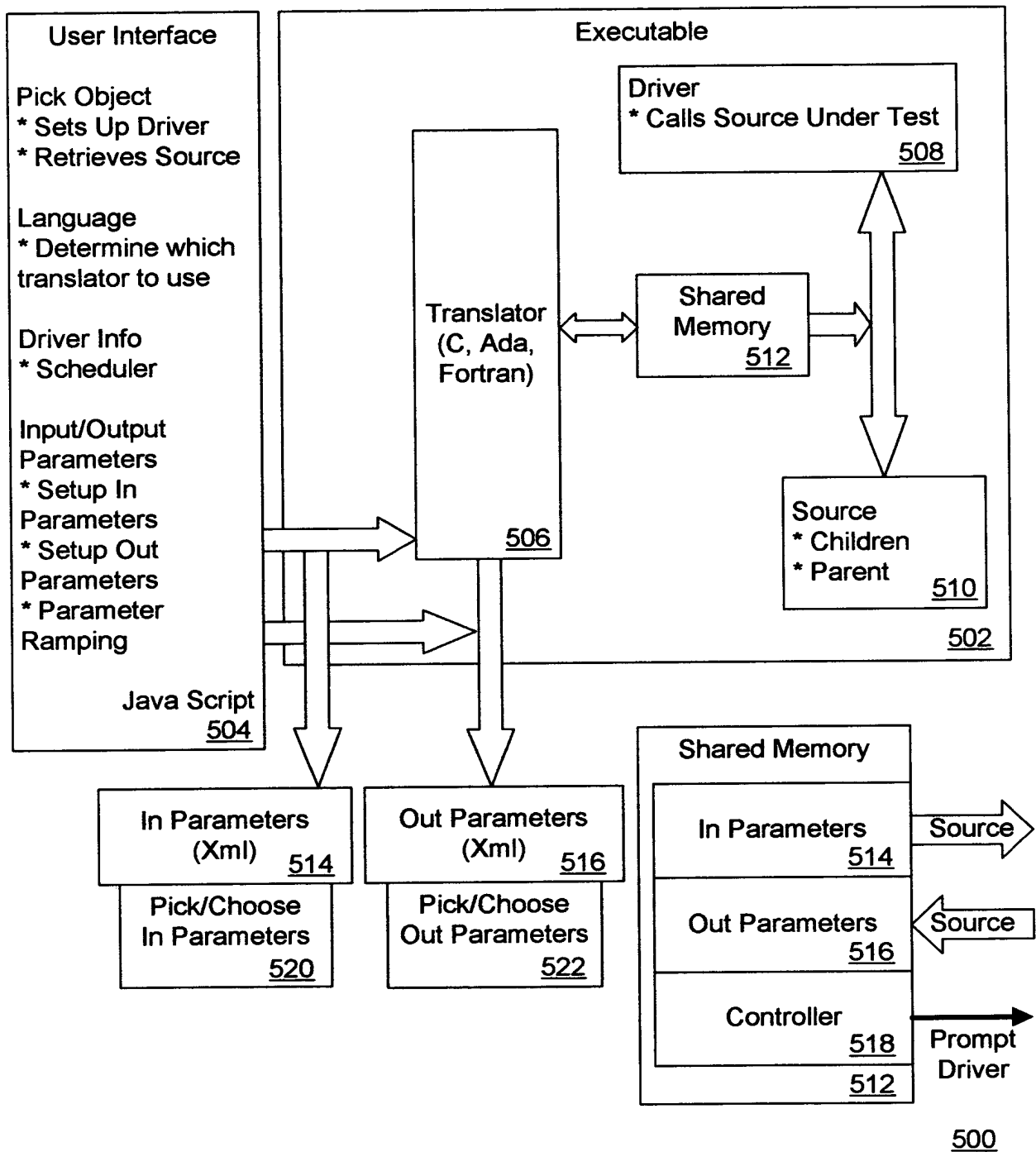


FIG. 5

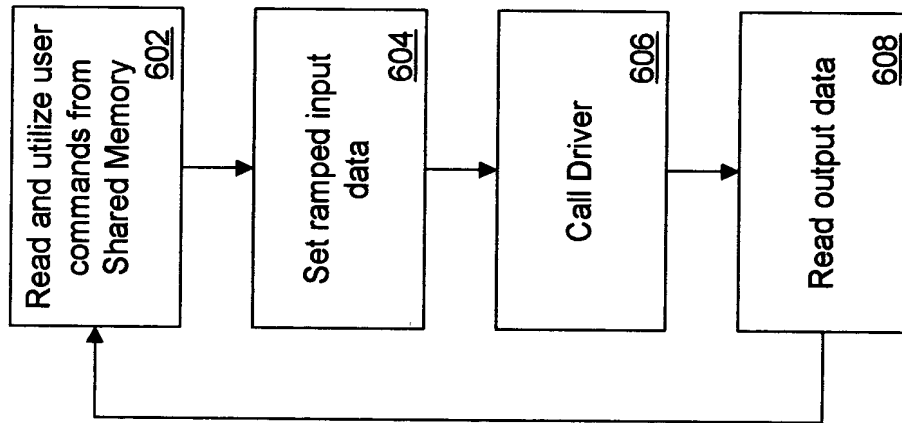


FIG. 6

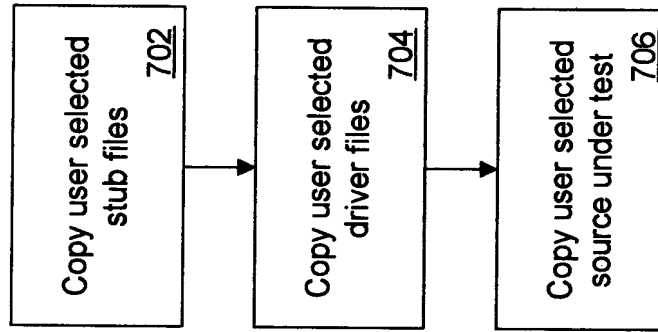


FIG. 7

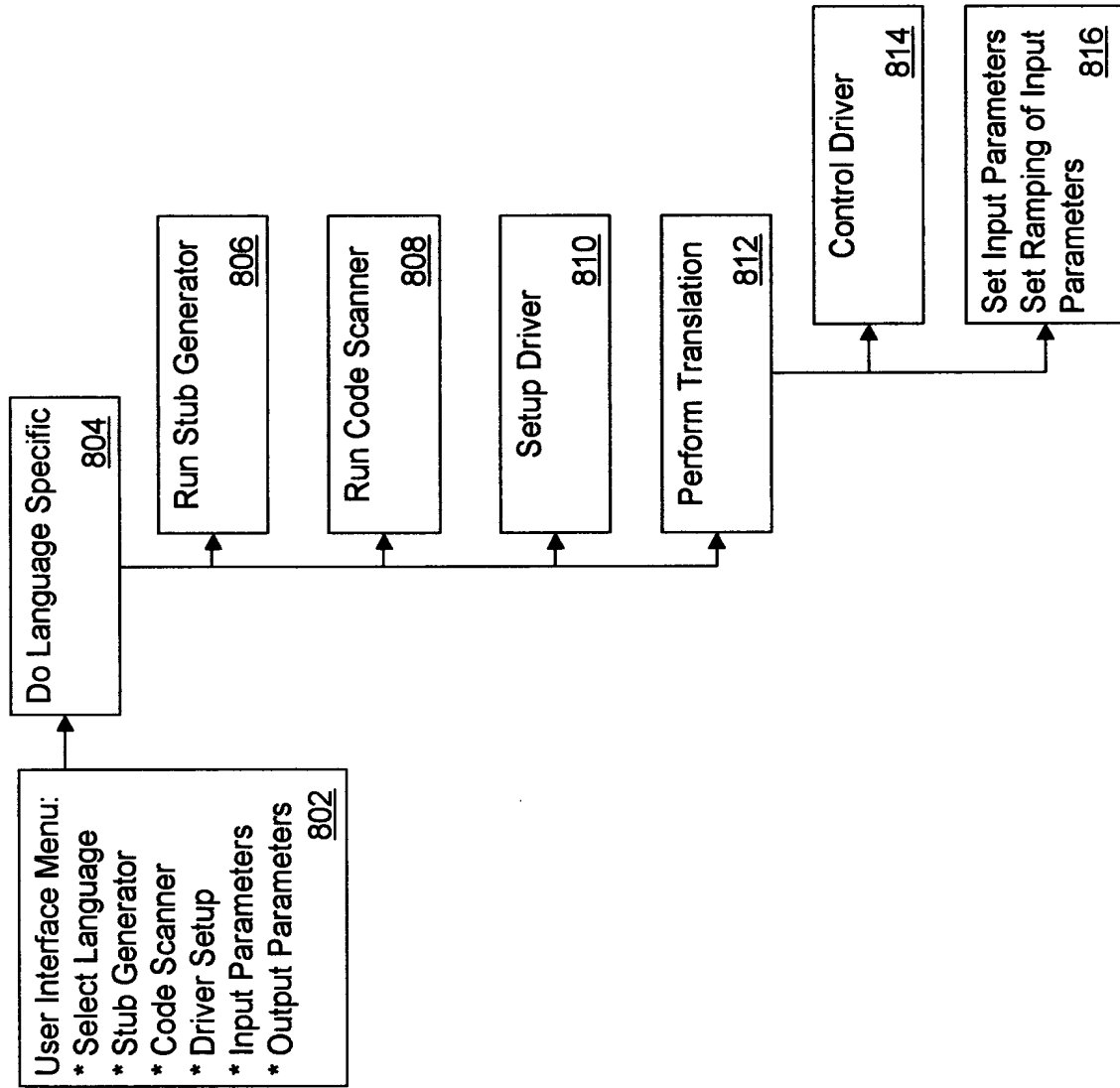


FIG. 8

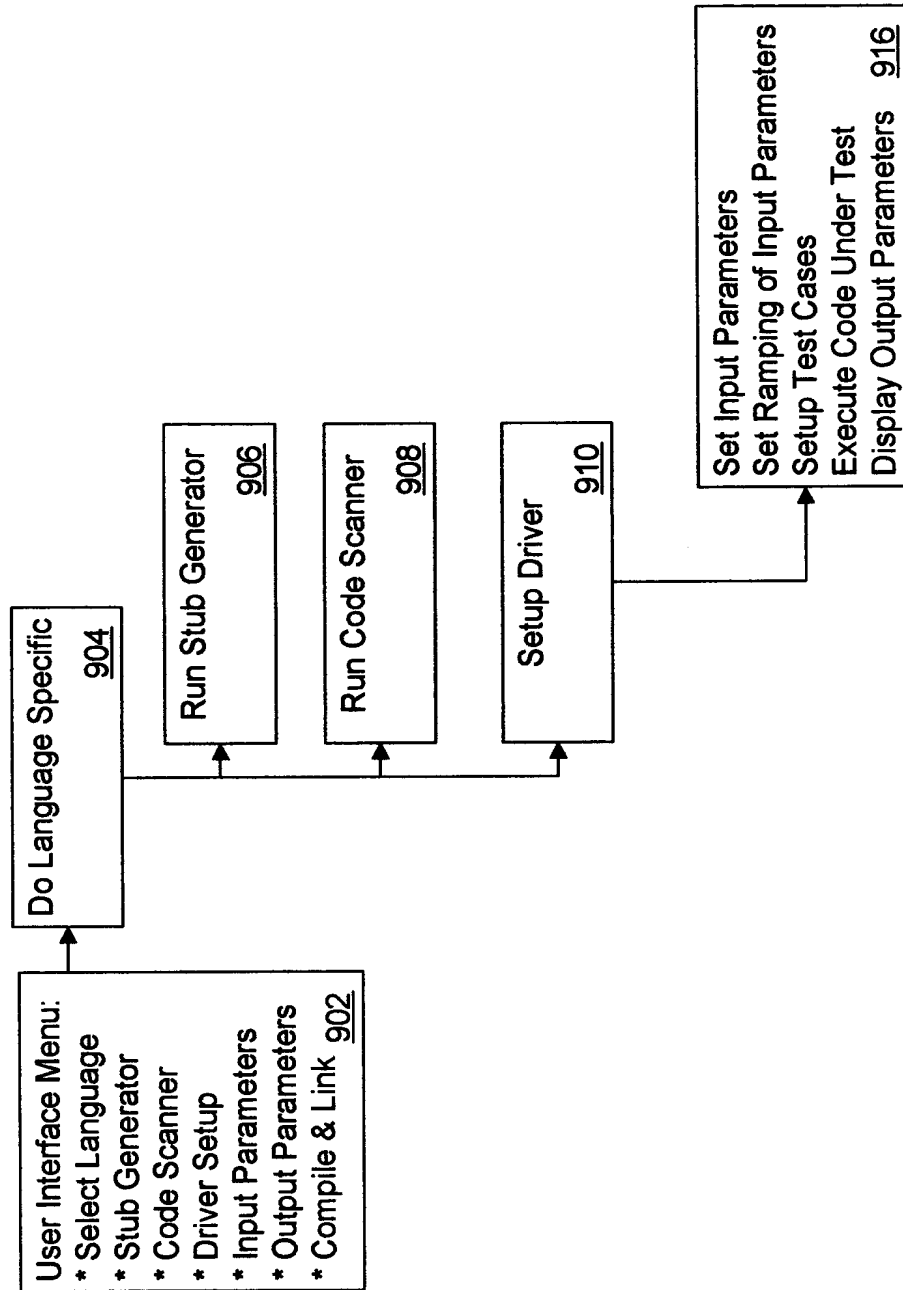


FIG. 9

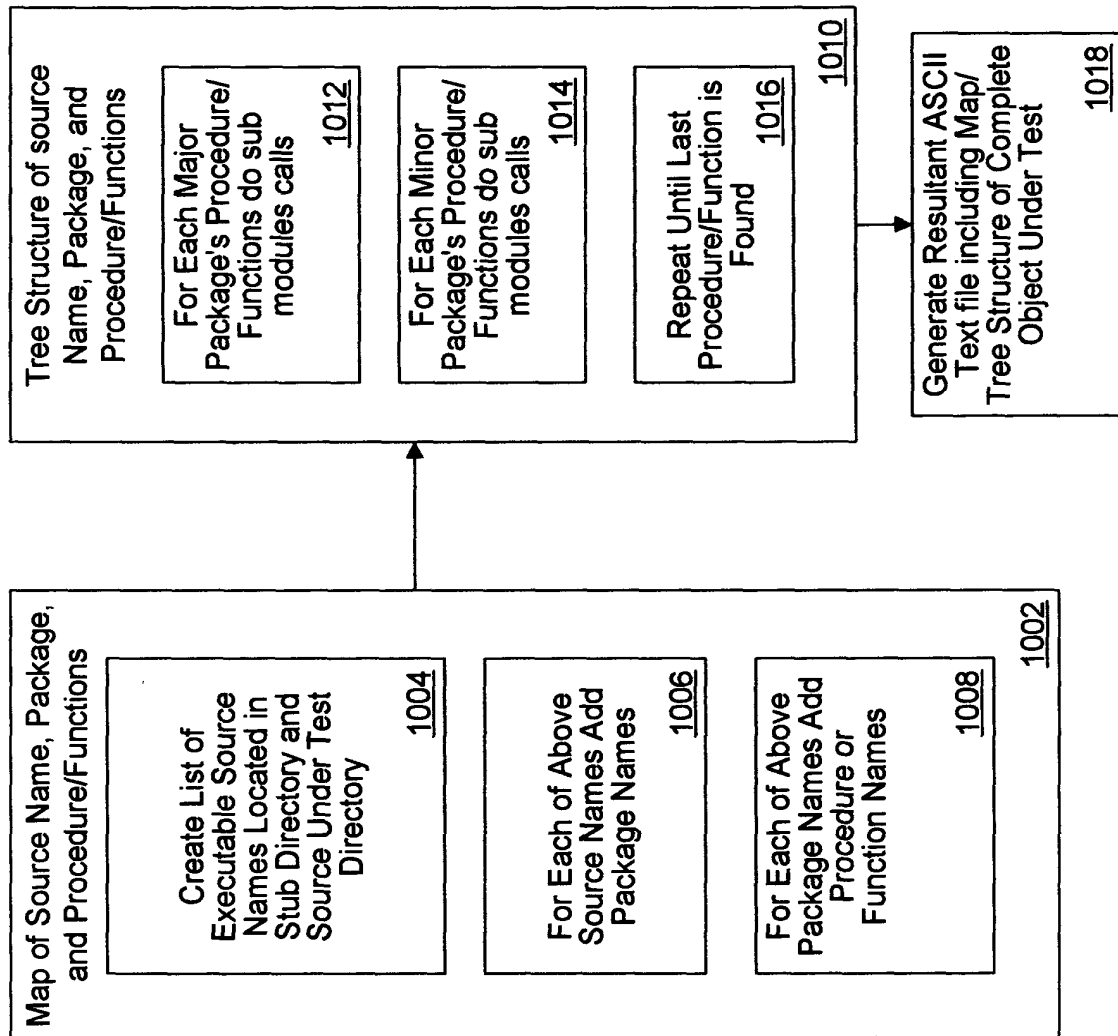


FIG. 10

10/29

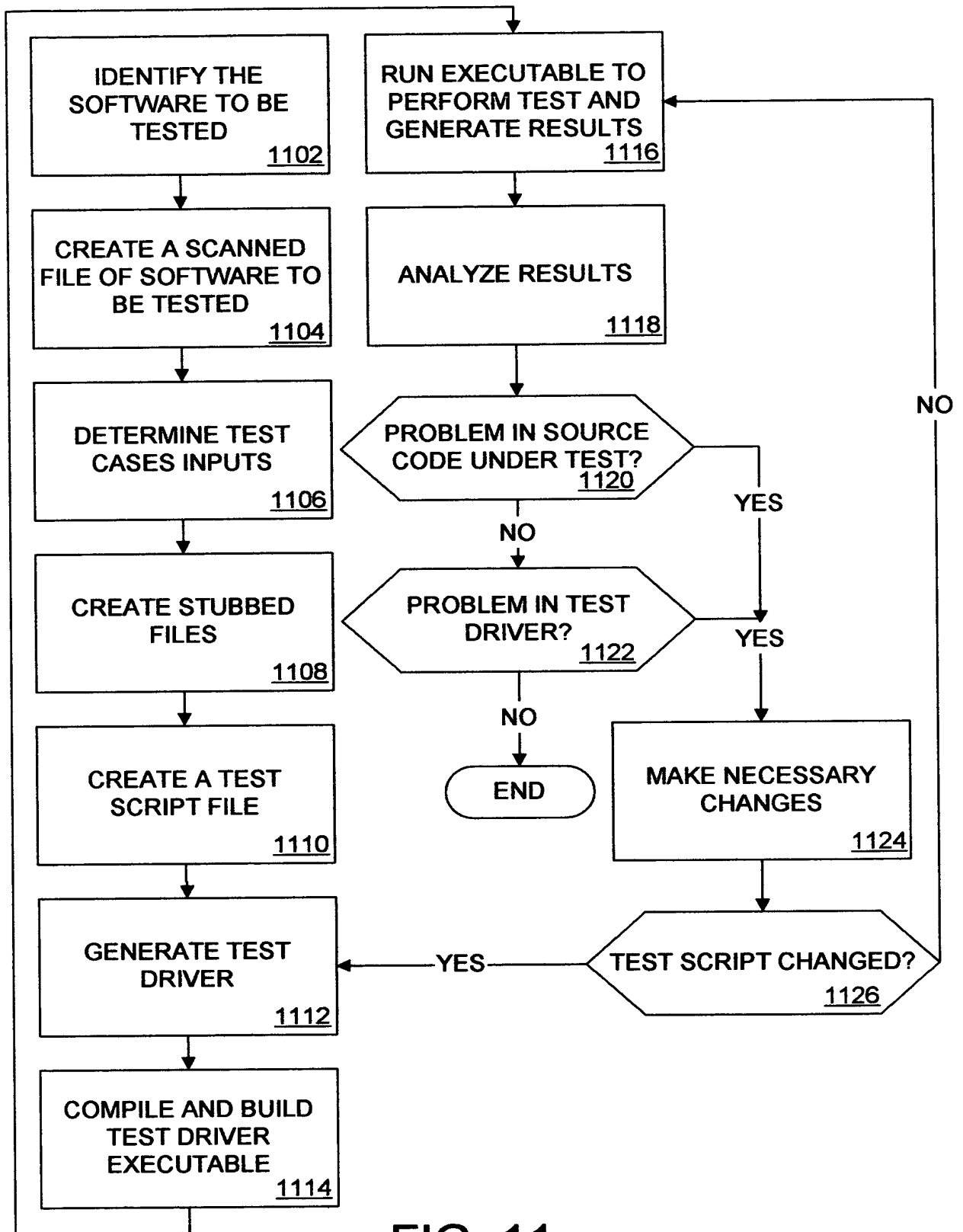


FIG. 11

 --Test Suite Setup File

--This file includes setup parameters for the test suite and can
 --be placed at C:\test_suite\
 -- ***** Note: Do not change the working directory path!!! *****
 --
 -- Specifies Which Editor to utilize
 -- (example: editor=notepad.exe)
 -- Note: If a editor does not exist, enter none
 -- (example: editor=none)
 --
 editor=notepad.exe
 --
 --PDL Delimiters
 -- Note: If no PDL enter: none
 -- Example: high_level=none
 -- Example: low_level=none
 --
 high_level=%%
 low_level=%
 --
 --Specify Directories
 --
 cm_source_dir=C:\test_suite\cm_source\
 ts_output_dir=C:\test_suite\ts_output\
 build_dir=C:\test_suite\build\
 working_dir=C:\test_suite\
 temp_dir=C:\test_suite\temp\
 --
 --Ada Parameters
 -- Note: ada_versions = ada83 or ada95
 --
 ada_version=ada95
 ada_spec_suffix=.ads
 ada_body_suffix=.adb
 ada_separate_suffix=.a
 ada_compiler=adagide.exe
 -- Note: If a ada compiler does not exist, enter none
 -- (example: ada_compiler=none)
 --
 --C Parameters
 --
 c_suffix=.c
 c_header_suffix=.h
 c_compiler=none
 -- Note: If a c compiler does not exist, enter none
 -- (example: c_compiler=none)
 --
 --Fortran Parameters
 --
 fortran_suffix=.f
 fortran_compiler=none
 -- Note: If a fortran compiler does not exist, enter none
 -- (example: fortran_compiler=none)
 --
 --Options *** Note: DO NOT MODIFY ***
 date_stamp=0
 source_code_rev=0
 test_comments=1
 spec_refs=1
 auto_seq=1

FIG. 12

12/29

```
SOURCE_UNDER_TEST=radar_altimeter.adb
SOURCE_VERSION=
SCCS INFO: @(#) radar_altimeter.adb 1.26 01/09/18
10:47:03
PACKAGE=Radar_Altimeter
START_VECTOR_DATA
  OUTPUT_VARIABLE_F=Analog_Input.Lt_Antenna_Volts
  EXPECTED_VALUE=0.0
  TOLERANCE_P=0.0
END_VECTOR_DATA
PROCEDURE/FUNCTION=Update
START_TEST=1
START_COMMENT
  update - Ramp antenna voltage up to 6 volts. Compute
           the radar altitude, Rate of climb and terrain
           slope for each iteration. Radar altitude will increase on
           each iteration and the ROC will remain constant.
END_COMMENT
START_REQUIREMENT
  The Radar Altimeter processing will select the appropriate transmit and
  receive antenna pair based on the aircraft attitude to provide radar
  altimeter operation over the required attitude range. Radar Altimeter
  processing shall provide continuous coverage below 5,000 feet AGL.
  [SRS-459]
END_REQUIREMENT
NUM_ITERATIONS=1
START_ITER=1
  START_INPUT_CODE
    with Common_Types;
    Volts : Common_Types.Float_32_Type;
    with Navigation;
    Navigation.Inertial_Velocity(Navigation.North) := 500.0;
    Navigation.Inertial_Velocity(Navigation.Down) := 26.0;
    Volts := 8.0;
    with Analog_Input;
    Analog_Input.Put_Antenna_Volts(Analog_Input.Lt_Ant, Volts);
    Analog_Input.Put_Antenna_Volts(Analog_Input.Rt_Ant, Volts);
    Analog_Input.Put_Antenna_Volts(Analog_Input.Ctr_Ant, Volts);
    with Antenna;
    Antenna.Update;
    Navigation.Roll_Angle.Value := 45.0;
    Navigation.Roll_Angle.Valid := True;
    with Attitude_Alt;
    Attitude_Alt.Roll_Angle.Value := 0.0;
    Attitude_Alt.Roll_Angle.Valid := False;
  END_INPUT_CODE
  START_OUTPUT_CODE
    with Common_Types;
    Radar_Altitude : Common_Types.Validity_Test_Type;
    with Radar_Altimeter;
    Radar_Altitude := Radar_Altimeter.Get_Radar_Altitude;
  END_OUTPUT_CODE
  START_EXPECTED_RESULTS
    OUTPUT_VARIABLE_F=Radar_Altitude.Value
    EXPECTED_VALUE=120.25
    TOLERANCE_R=0.0
  END_EXPECTED_RESULTS
  START_EXPECTED_RESULTS
    OUTPUT_VARIABLE_B=Radar_Altitude.Valid
    EXPECTED_VALUE=TRUE
    TOLERANCE_P=N/A
  END_EXPECTED_RESULTS
END_ITER=1
END_TEST=1
END_TEST_FILE
```

1300

FIG. 13

Created On: 07/16/2002 13:53:00

** SCCS INFO: @(#) radar_altimeter.adb 1.26 01/09/18 10:47:03

**Requirement Comments:

** The Radar Altimeter processing will select the appropriate transmit and receive antenna pair based on the aircraft attitude to provide radar altimeter operation over the required attitude range. Radar Altimeter processing shall provide continuous coverage below 5,000 feet AGL.

** [SRS-459]

**Test Description Comments:

** update - Ramp antenna voltage up to 6 volts. Compute the radar altitude, Rate of climb and terrain slope for each iteration. Radar altitude will increase on each iteration and the ROC will remain constant.

**Package Name: Radar_Altimeter

**Procedure Name: Update

**Test Case Number: 1

Variable Name	Actual Value	Expected Value	Tolerance
Results			
Radar_Altitude.Value	1.20250E+02	1.20250E+02 (+/-) 0.00000E+00	
PASS			
Radar_Altitude.Valid	TRUE	TRUE	N/A PASS

Vector Data Results

Analog_Input.Lt_Antenna_Volts				
8.00000E+00	0.00000E+00	(%) 0.00000E+00	FAIL	

SUMMARY For: radar_altimeter_drv

Number Passed:	2
Number Failed:	1
Total:	3

1400

FIG. 14

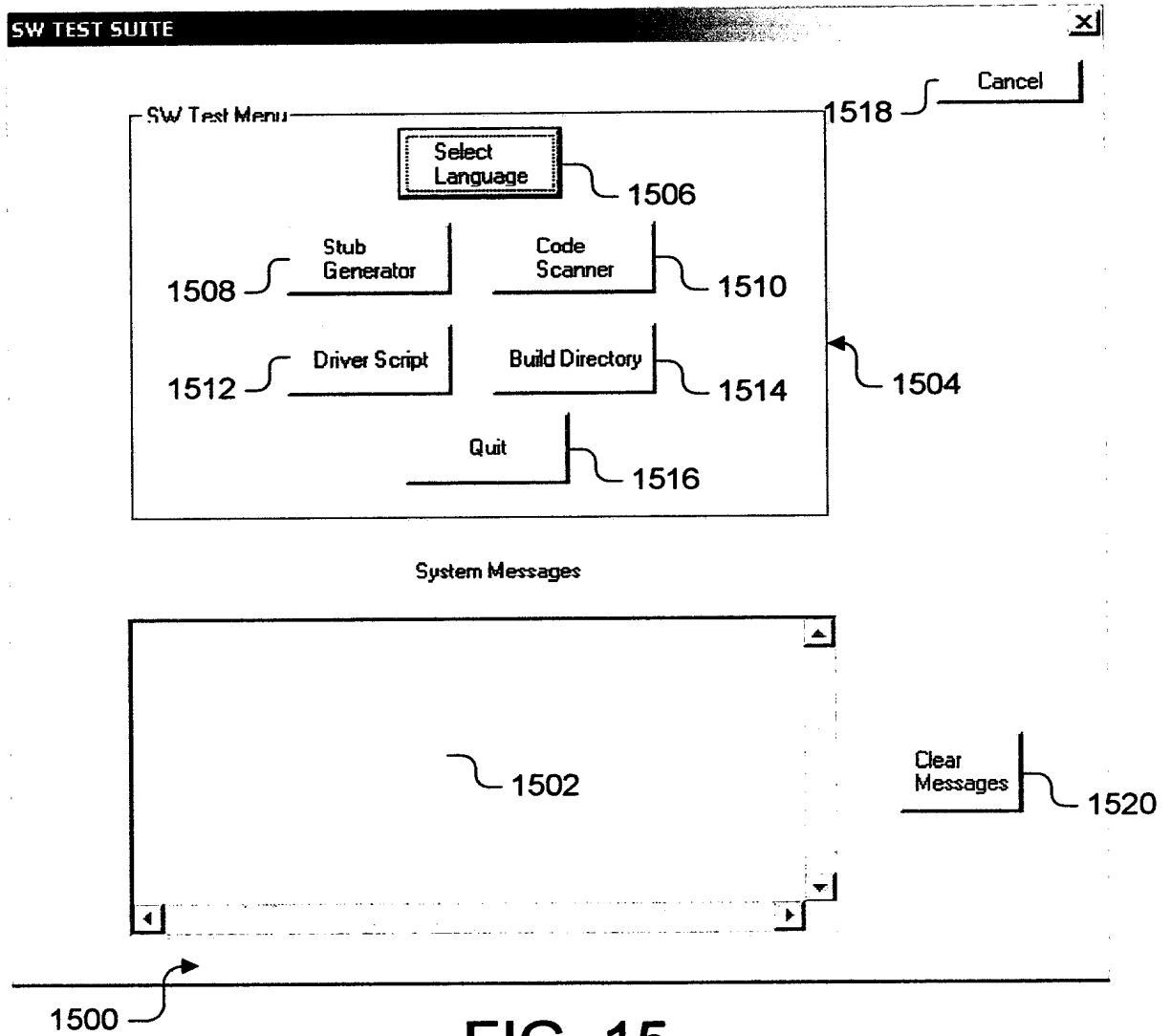


FIG. 15

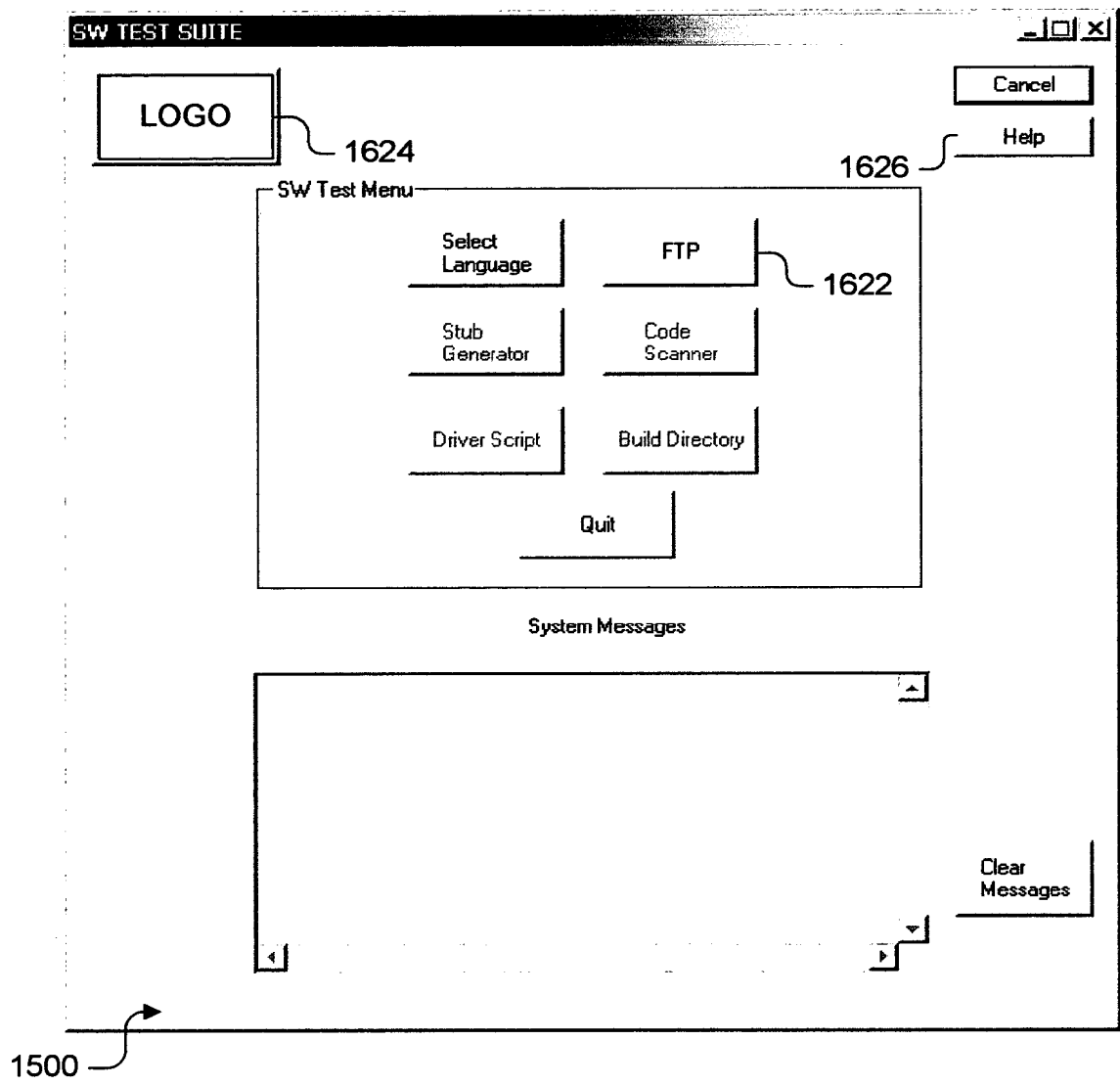


FIG. 16

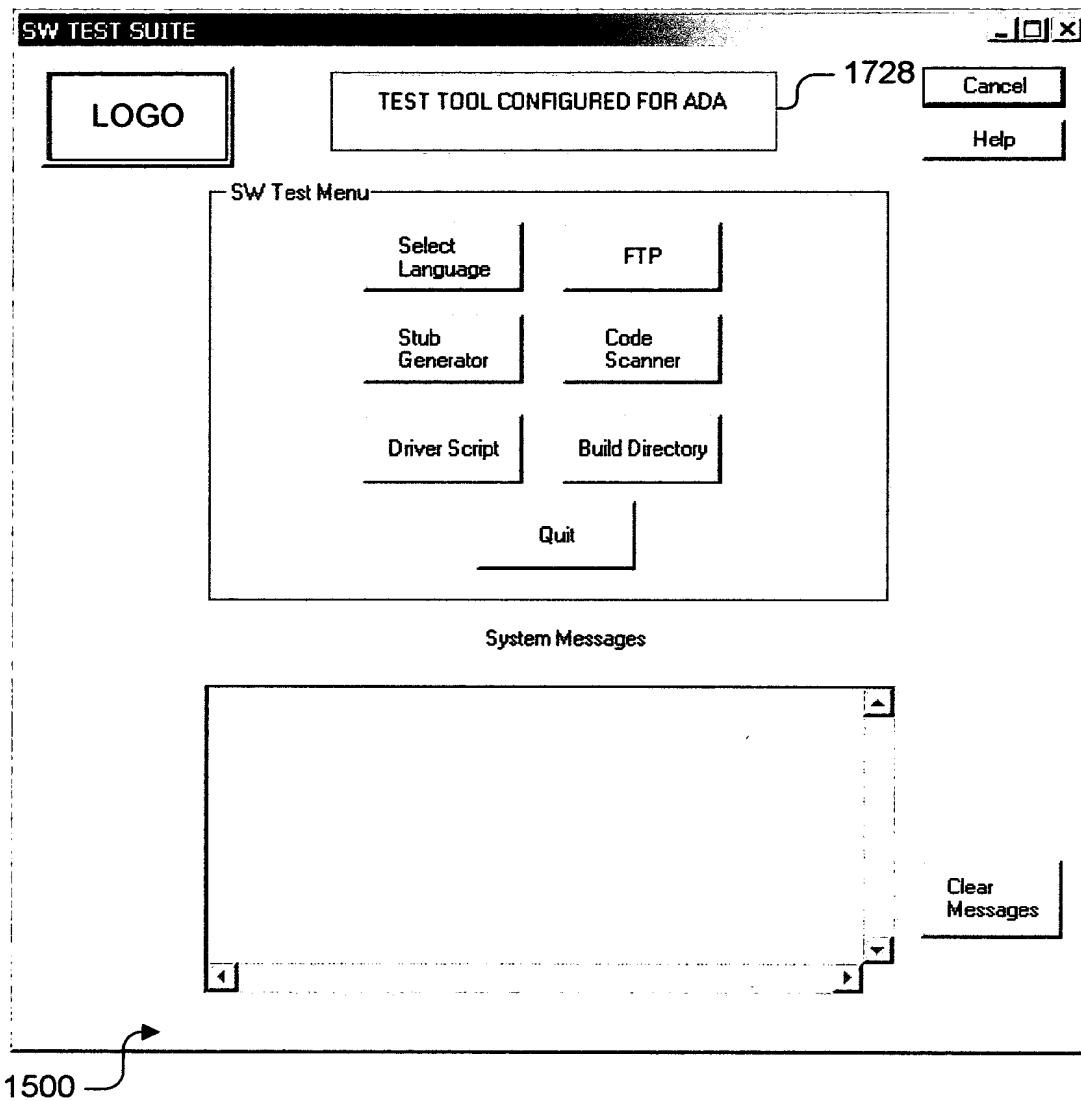


FIG. 17

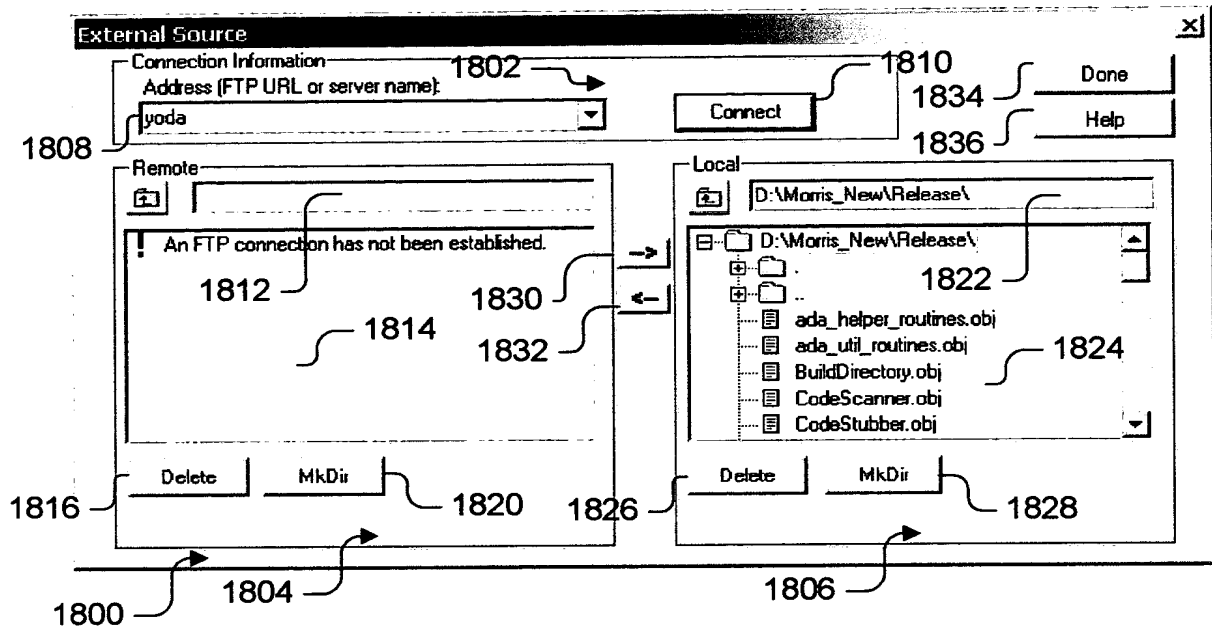


FIG. 18

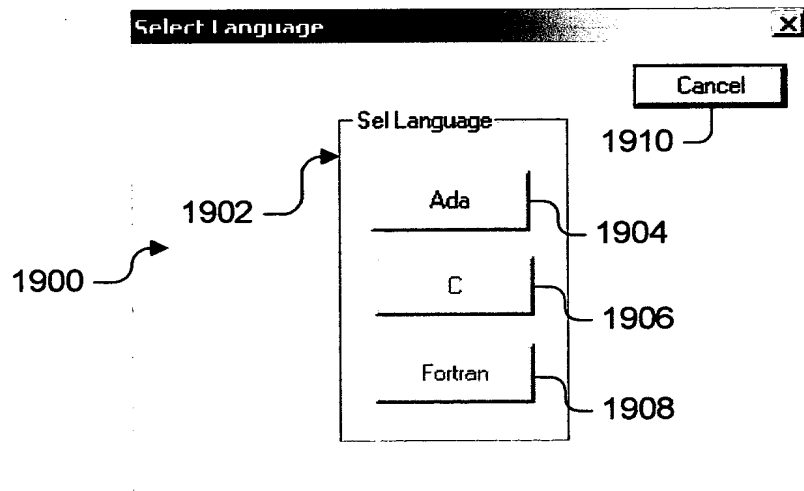


FIG. 19

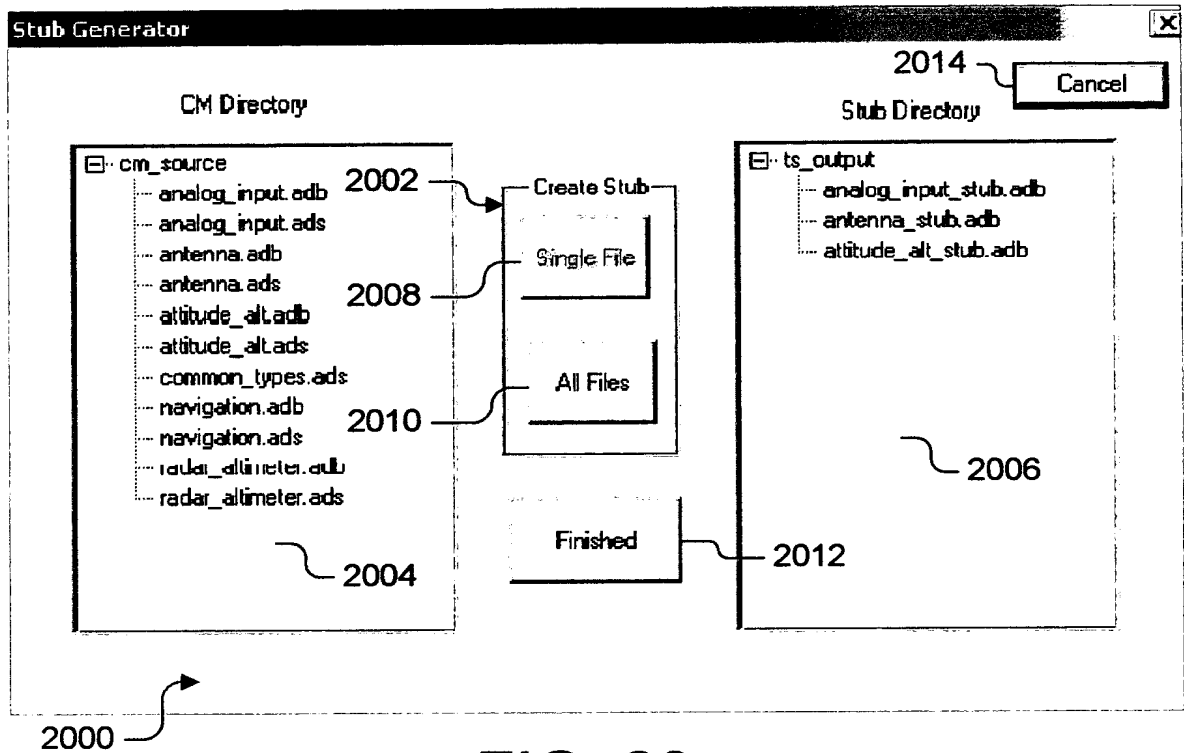


FIG. 20

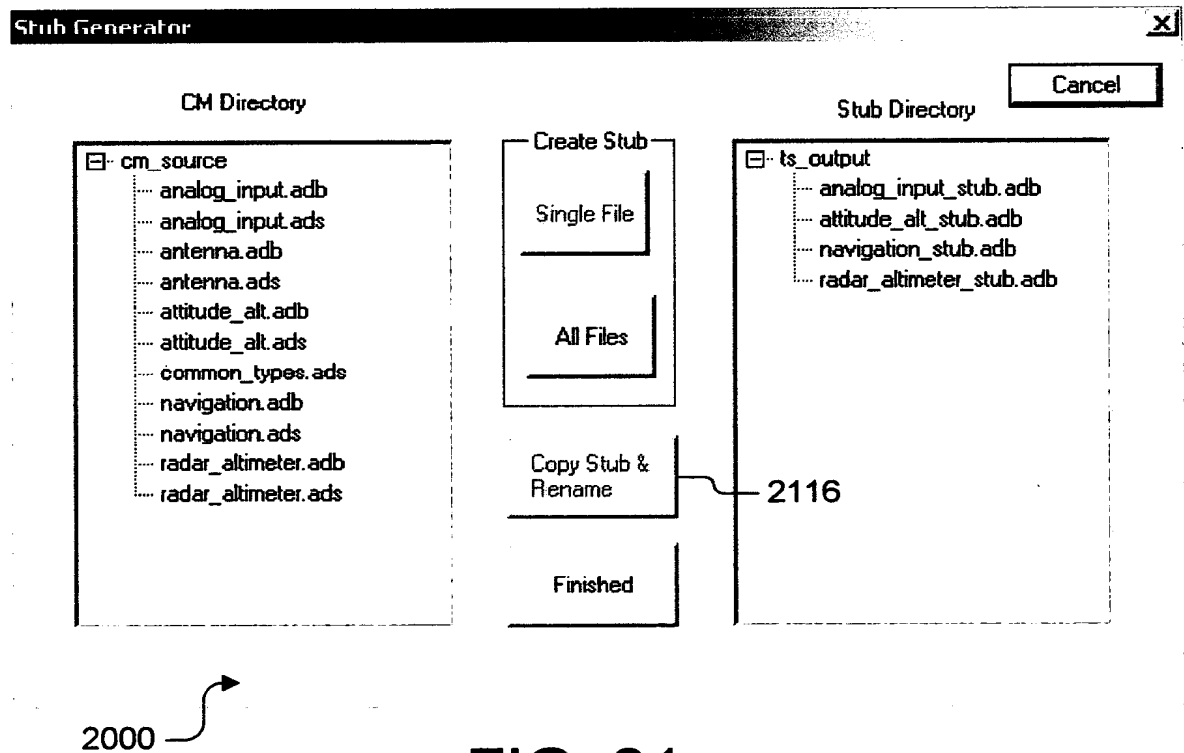


FIG. 21

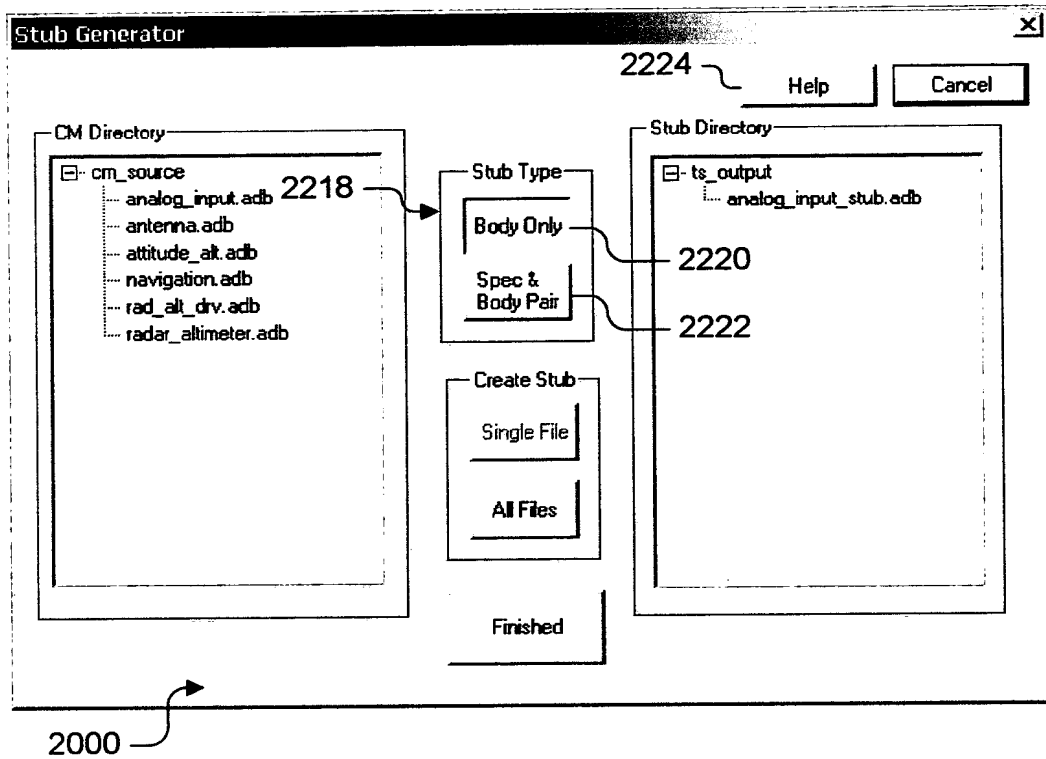


FIG. 22

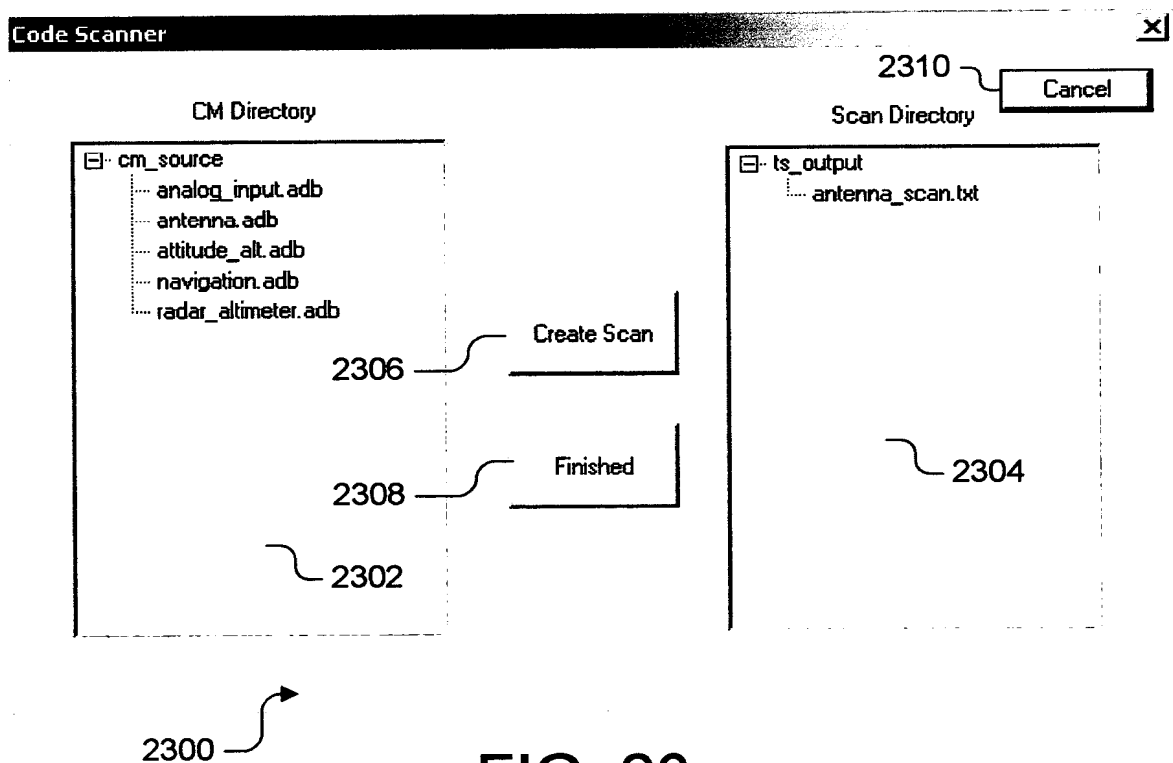


FIG. 23

```

Source File: analog_input.adb
package body Analog_Input is
  procedure Put_Antenna_Volts
    (Which_One : in Antenna_Type;
     Volts : in Common_Types.Float_32_Type) is
  case Which_One is
    when Lt_Ant =>
    when Rt_Ant =>
    when Ctr_Ant =>
    when others =>
  function Get_Left_Ant
    return Common_Types.Float_32_Type is
  function Get_Right_Ant
    return Common_Types.Float_32_Type is
  function Get_Ctr_Ant
    return Common_Types.Float_32_Type is
end Analog_Input;

```

```

Number of Packages = 1
Number of Procedures = 1
Number of Functions = 3
Lines of PDL = 0
Lines of Code (LOC) = 15

```

2400

FIG. 24

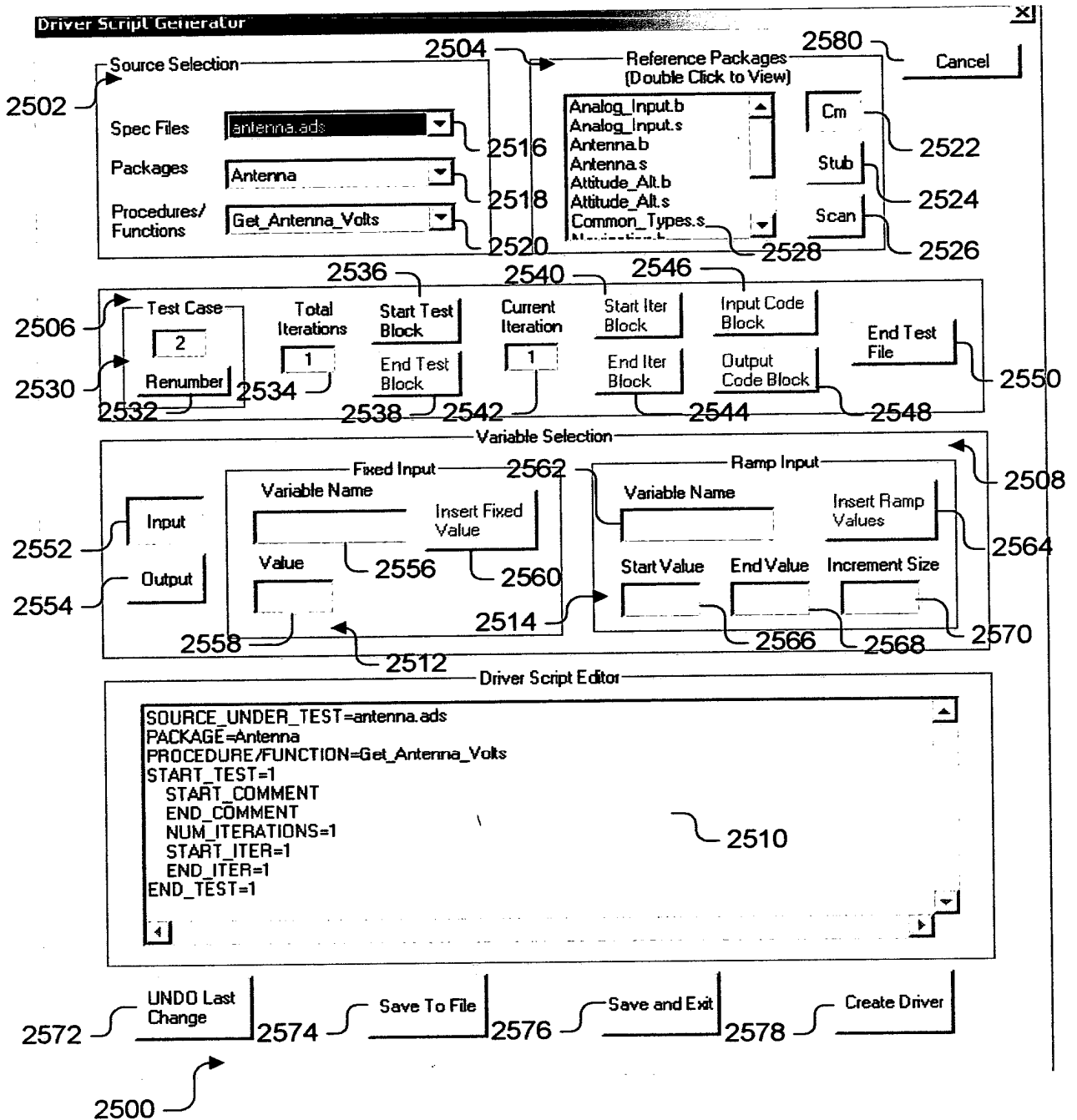


FIG. 25

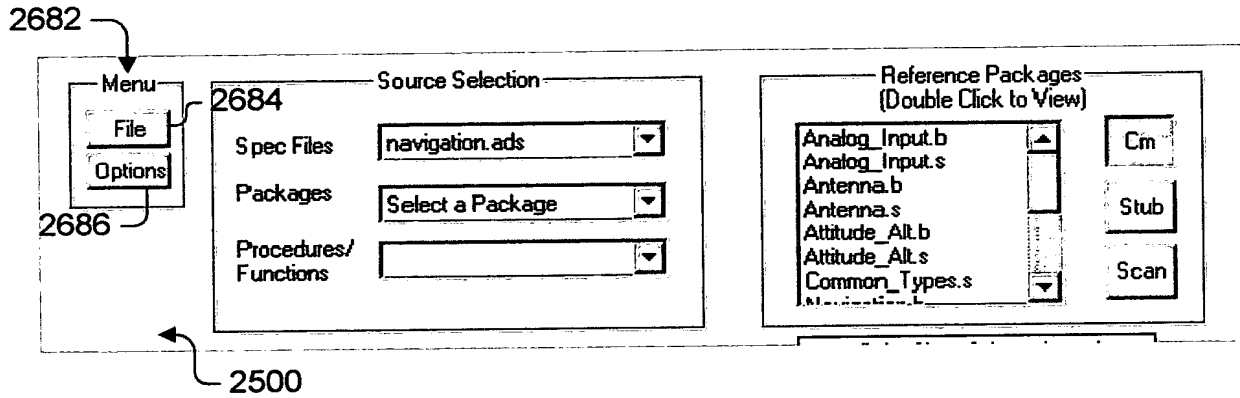


FIG. 26

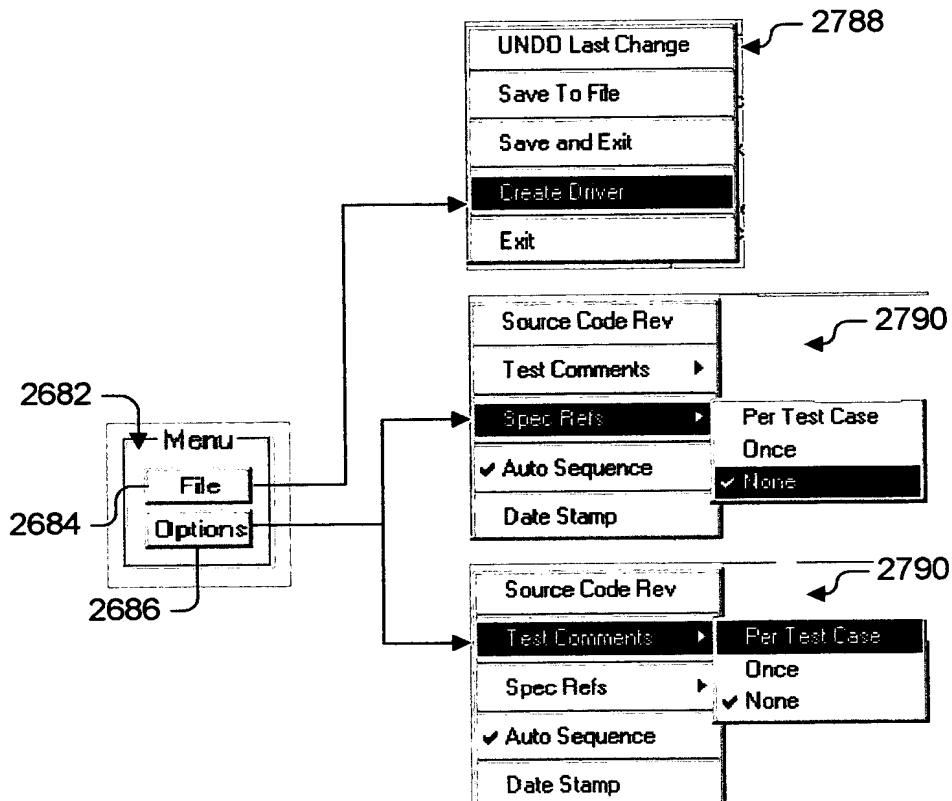


FIG. 27

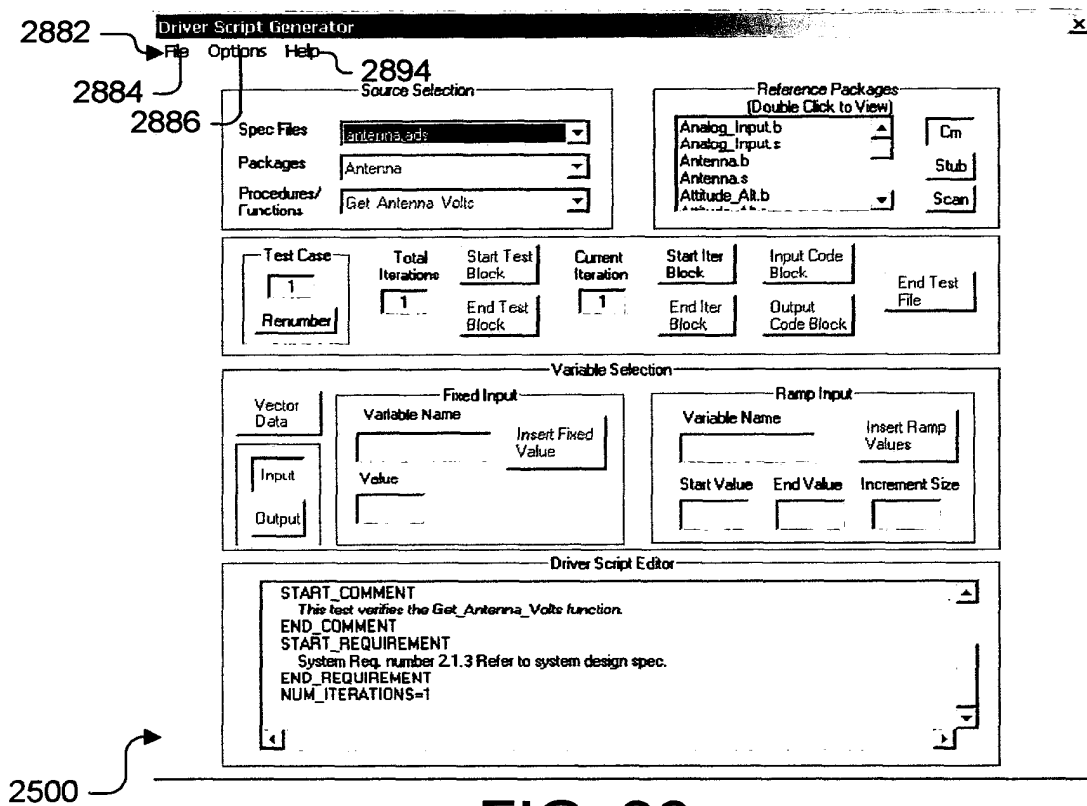


FIG. 28

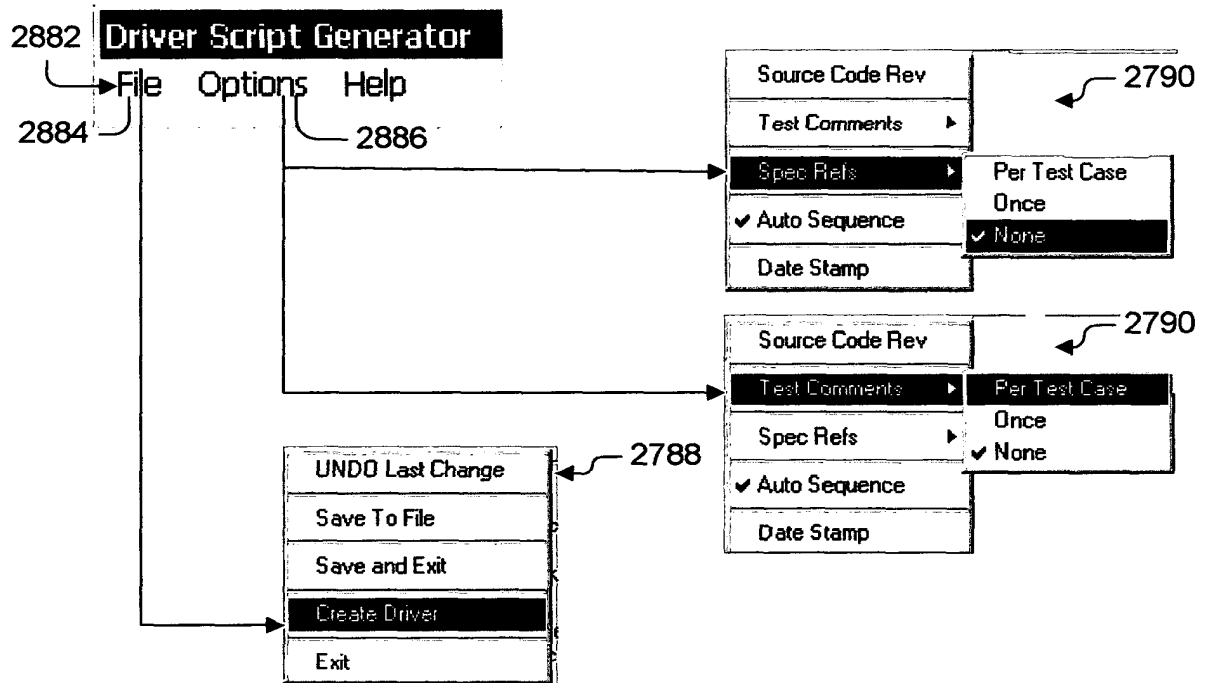


FIG. 29

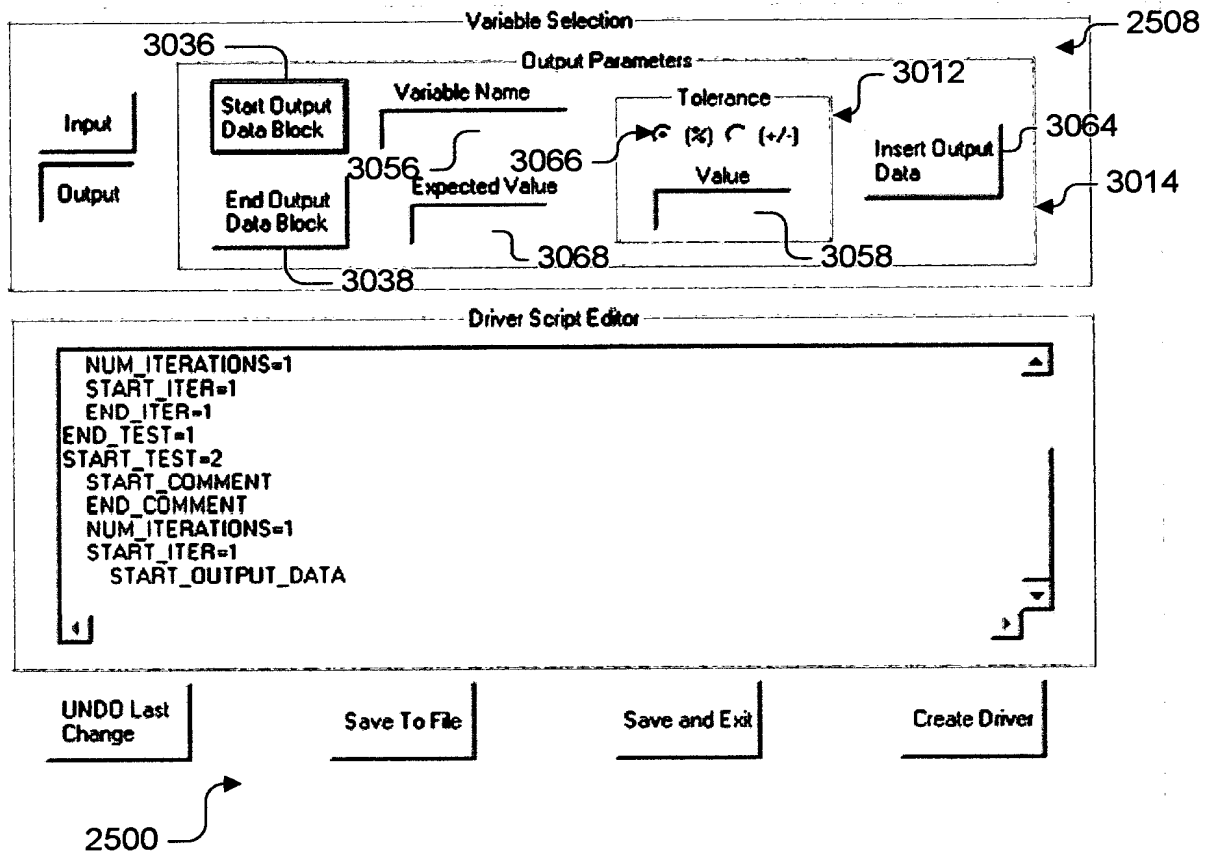


FIG. 30

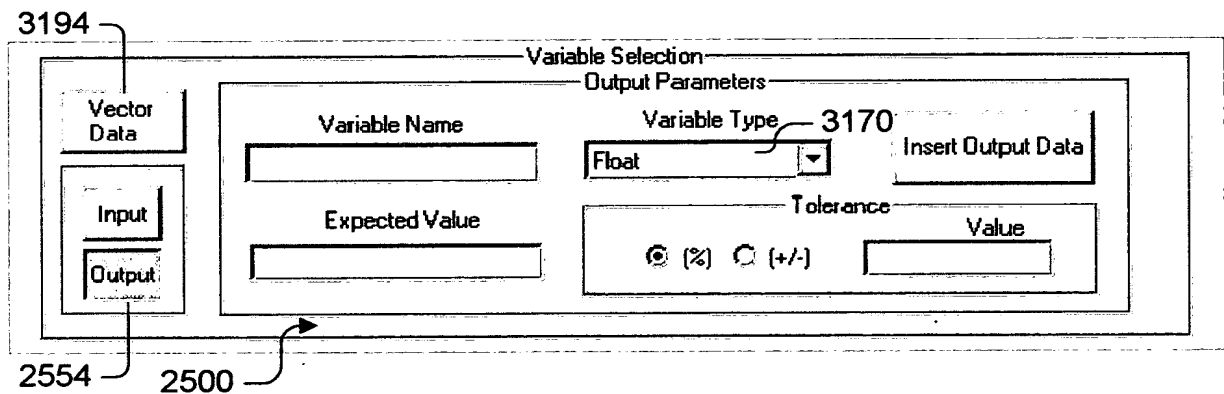


FIG. 31A

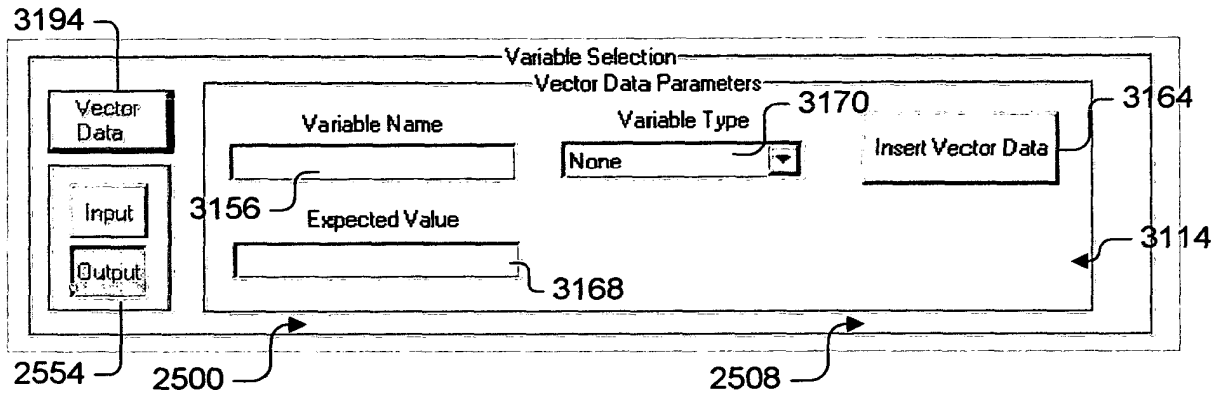


FIG. 31B

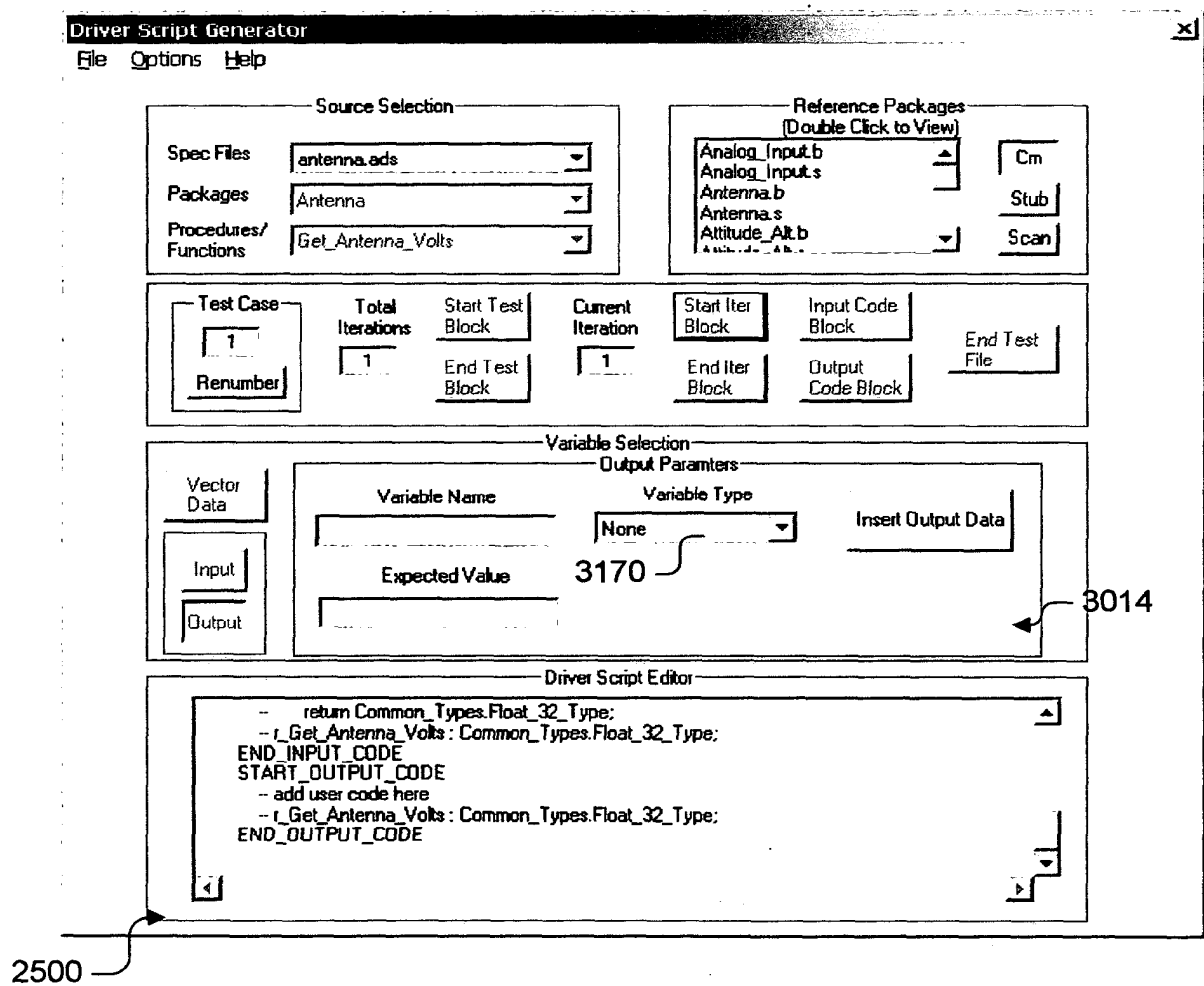


FIG. 32

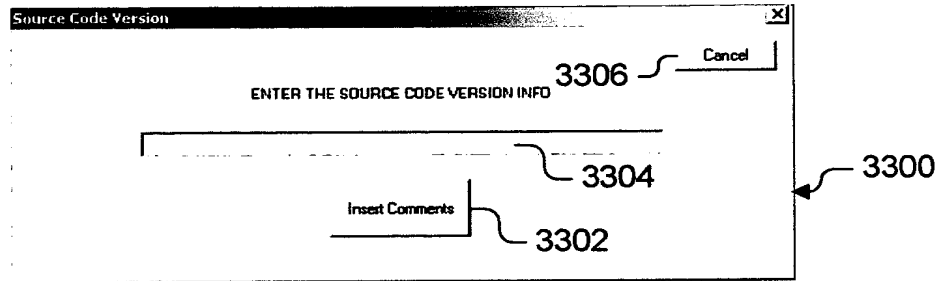


FIG. 33

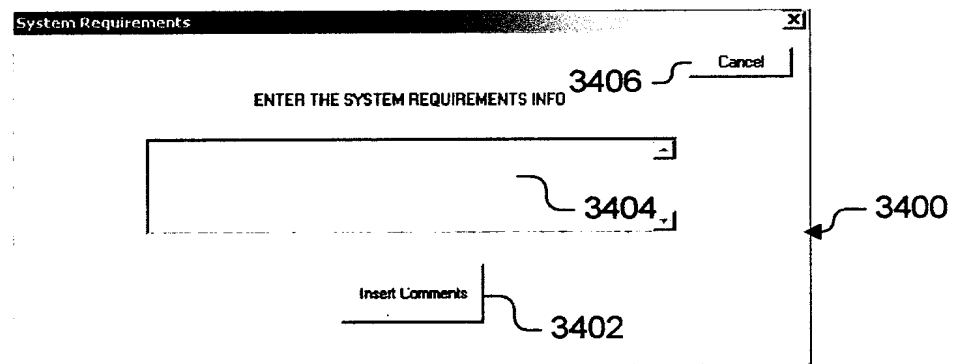


FIG. 34

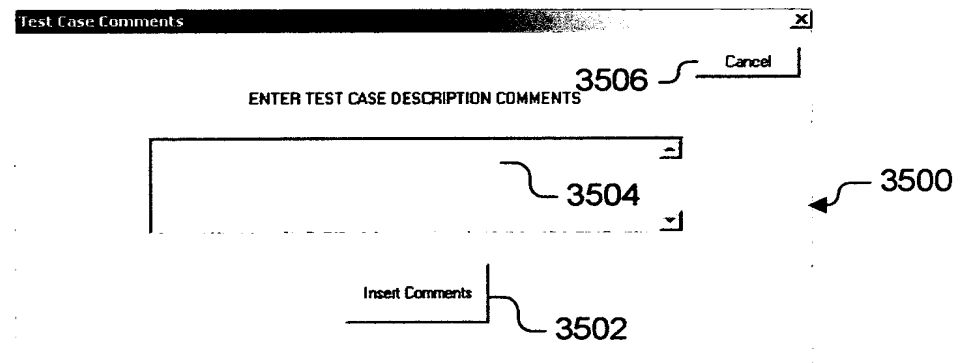


FIG. 35

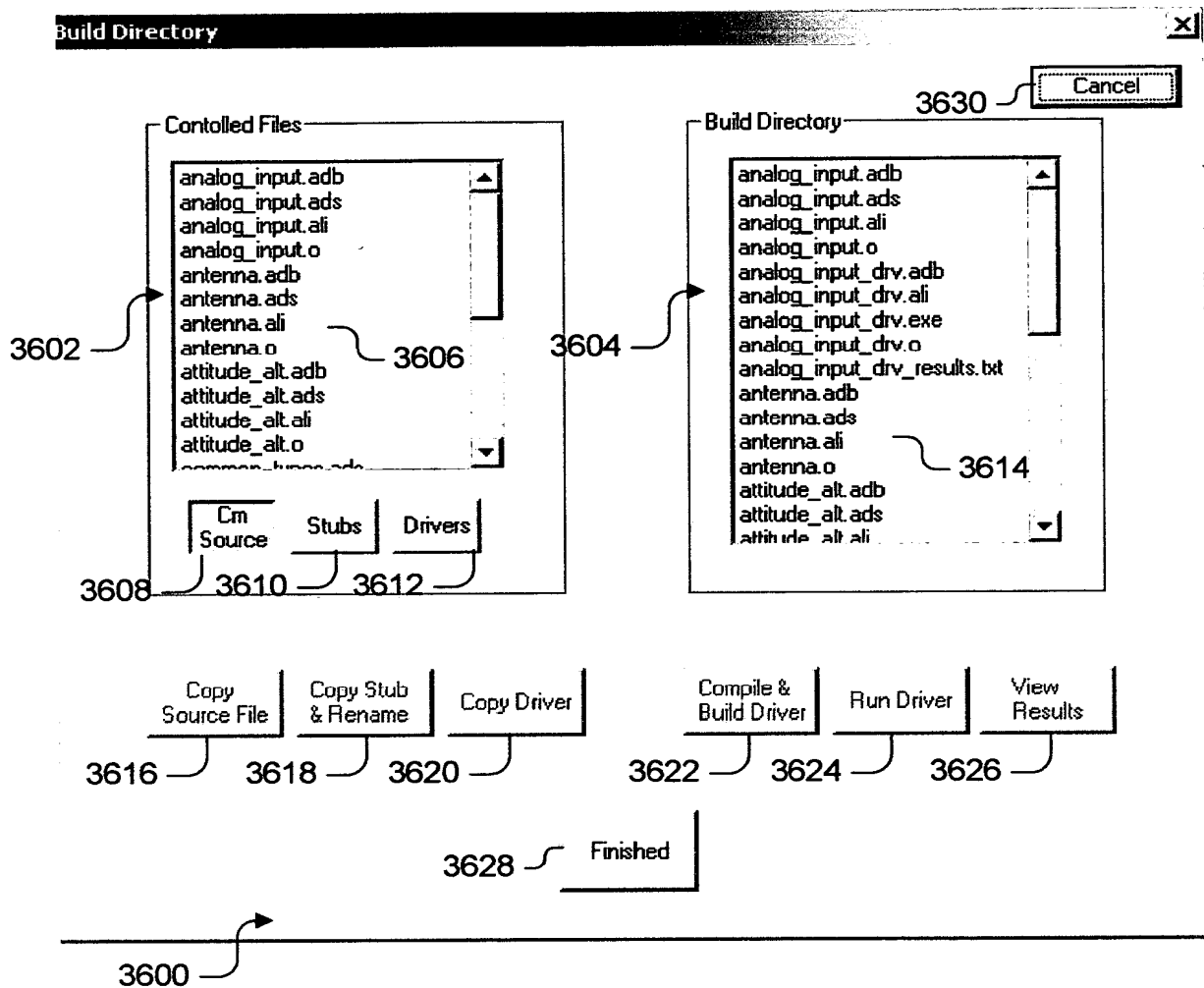


FIG. 36

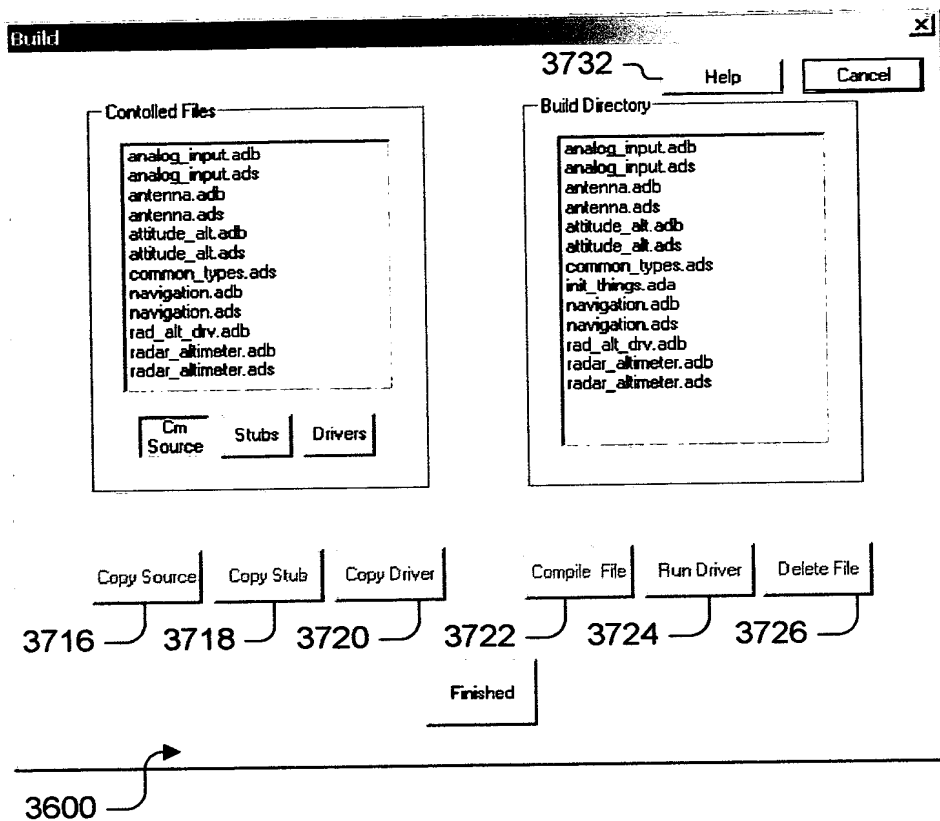


FIG. 37

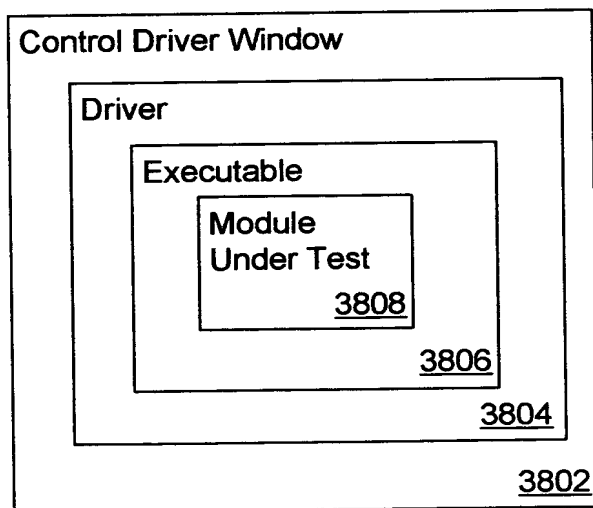


FIG. 38

3800

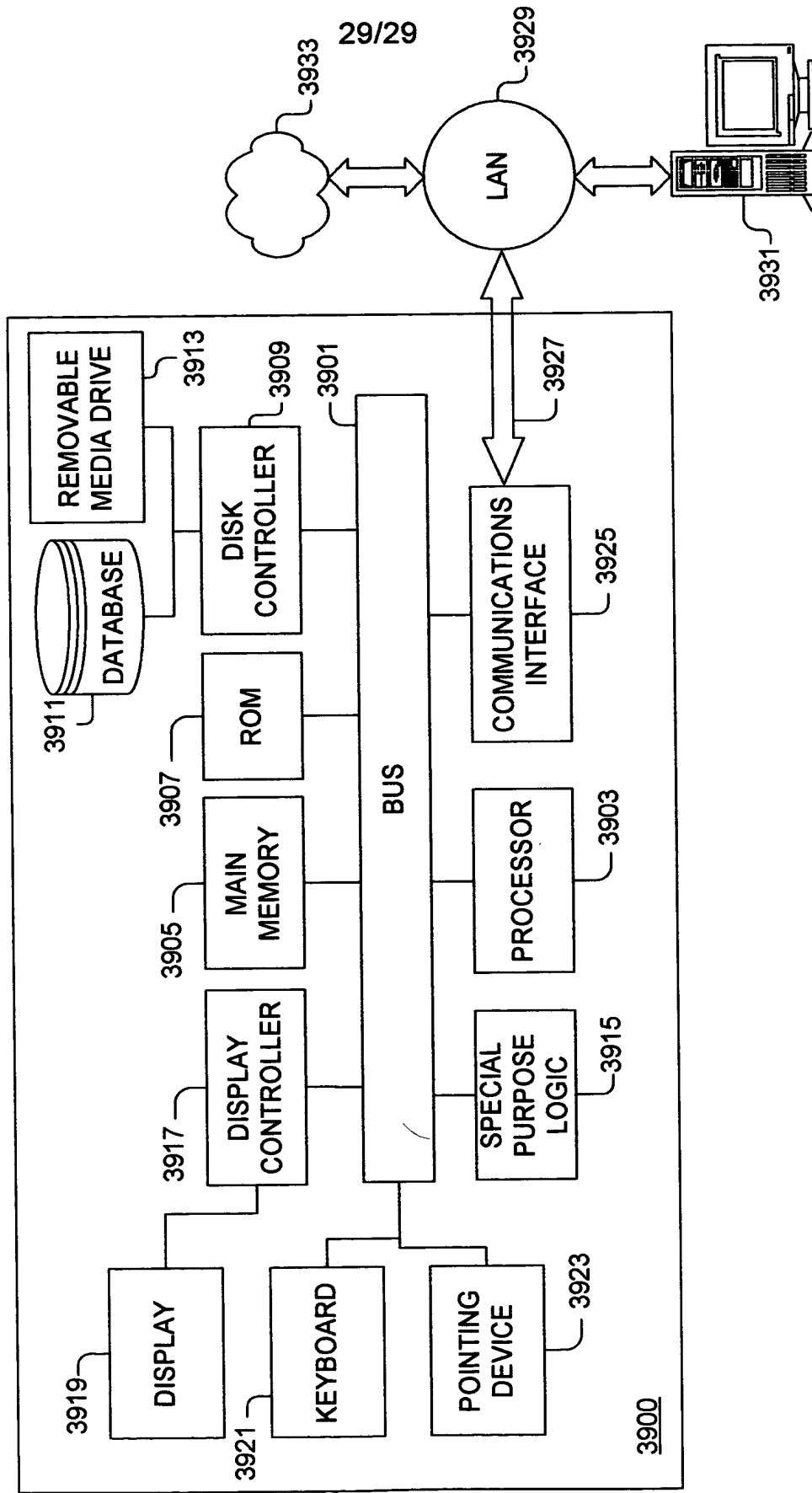


FIG. 39